

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listings of claims in the above-referenced application.

Listing of Claims:

1. (Currently Amended) A method of using a local storage device to read desired data while the data is being ~~stored on~~ transferred from the local storage device to a remote storage device using the cache of the local storage device in connection with transferring chunks of data from the local storage device to the remote storage device, the method comprising:

if the desired data is entirely in a cache of the local storage device, the local storage device returning the data from the cache, wherein the cache of the local storage device contains data written to the local storage device begun after a first time and before a second time that is associated with a first chunk of data and contains data written to the local storage device begun after the second time that is associated with a second chunk of data different from the first chunk of data and wherein after completion of all writes associated with the first chunk of data, the local storage device initiates transfer of writes associated with the first chunk of data to the remote storage device, an order of the transfer being independent of an order in which data writes of the first chunk are provided to the local storage device; and

if the desired data is not entirely in the cache of the local storage device, reading data ~~for parts of a particular track~~ from the remote storage device to the local storage device and the local storage device merging the data ~~for the parts of the particular track~~ from the remote storage device with data ~~corresponding to different parts of the particular track~~ from the cache of the local storage device at the local storage device ~~to interleave the parts for the particular track from the remote storage device with the different parts for the particular track from the local storage device.~~

2. (Previously Presented) A method, according to claim 1, further comprising:

prior to reading data from the remote storage device to the local storage device, creating a temporary storage area at the local storage device if there is data from the local storage device that is to be used.

3. (Original) A method, according to claim 2, wherein the temporary storage area is a scratch slot.

4. (Previously Presented) A method, according to claim 3, further comprising:

prior to creating a temporary storage area, locking slots of the local storage device that correspond to data from the local storage device that is to be used.

5. (Previously Presented) A method, according to claim 4, further comprising:

after merging the data, unlocking the slots of the local storage device that correspond to data from the local storage device that is to be used.

6. (Original) A method, according to claim 1, wherein the data from the local storage area is merged on top of data from the remote storage area.

7. (Original) A method, according to claim 1, further comprising:

the remote storage device allocating a temporary storage area in response to data to be read being stored in a cache slot of the remote storage device.

8. (Original) A method, according to claim 7, further comprising:

reading data from the disk of the remote storage area into the temporary storage area; and
merging the data to be read stored in the cache slot with data from a disk in the temporary storage area.

9. (Original) A method, according to claim 7, further comprising:

prior to the remote storage area determining if there is data to be read stored in a cache slot of the remote storage device, the remote storage device writing at least a portion of the data from at least one cache slot of the remote storage device to a disk of the remote storage device.

10. (Currently Amended) Computer software, stored in a computer-readable medium, that reads desired data while the data is being ~~stored on~~ transferred from the local storage device to a remote storage device using the cache of the local storage device in connection with transferring chunks of data from a local storage device to the remote storage device, comprising:

executable code that returns the data from the cache if the desired data is entirely in a cache of the local storage device, wherein the cache of the local storage device contains data written to the local storage device begun after a first time and before a second time that is associated with a first chunk of data and contains data written to the local storage device begun after the second time that is associated with a second chunk of data different from the first chunk of data and wherein after completion of all writes associated with the first chunk of data, the local storage device initiates transfer of writes associated with the first chunk of data to the remote storage device, an order of the transfer being independent of an order in which data writes of the first chunk are provided to the local storage device; and

executable code that reads data ~~for parts of a particular track~~ from the remote storage device to the local storage device and merges the data ~~for the parts of the particular track~~ from the remote storage device with data ~~corresponding to different parts of the particular track~~ from the cache of the local storage device at the local storage device if the desired data is not entirely in the cache of the local storage device ~~to interleave the parts for the particular track from the remote storage device with the different parts for the particular track from the local storage device.~~

11. (Previously Presented) Computer software, according to claim 10, further comprising:

executable code that creates a temporary storage area at the local storage device if there is data from the local storage device that is to be used prior to reading data from the remote storage device to the local storage device.

12. (Original) Computer software, according to claim 11, wherein the temporary storage area is a scratch slot.

13. (Previously Presented) Computer software, according to claim 12, further comprising:

executable code that locks slots of the local storage device that correspond to data from the local storage device that is to be used prior to creating a temporary storage area.

14. (Previously Presented) Computer software, according to claim 13, further comprising:

executable code that unlocks the slots of the local storage device that correspond to data from the local storage device that is to be used after merging the data.

15. (Original) Computer software, according to claim 10, wherein the data from the local storage area is merged on top of data from the remote storage area.

16. (Original) Computer software, according to claim 10, further comprising:

executable code that allocates a temporary storage area at the remote storage device in response to data to be read being stored in a cache slot of the remote storage device.

17. (Original) Computer software, according to claim 16, further comprising:

executable code that reads data from the disk of the remote storage area into the temporary storage area; and

executable code that merges the data to be read stored in the cache slot with data from a disk in the temporary storage area.

18. (Original) Computer software, according to claim 16, further comprising:

executable code that writes at least a portion of the data from at least one cache slot of the remote storage device to a disk of the remote storage device prior to the remote storage area determining if there is data to be read stored in a cache slot of the remote storage device.